# id: int spaceld: name: string

user: Vector<User> permission: binary visibility: binary

targetUsers: Vector<User>
messages: Vector<Message>

get y set correspondientes
addMessageToTablon(msg);

permission:
0-local read
1-local write
2-remote read
3-remote write

## visibility:

0-private (user group)

1-private (1 user)

2-public

3-super-public (anonymousUser)

### targetUsers será:

- -null cuando visibility sea 2 ó 3
- -vector de tamaño = 1 cuando visibility sea 1
- -vector de tamaño > 1 cuando visibility sea 0

### User

id: int name: surname1: surname2:

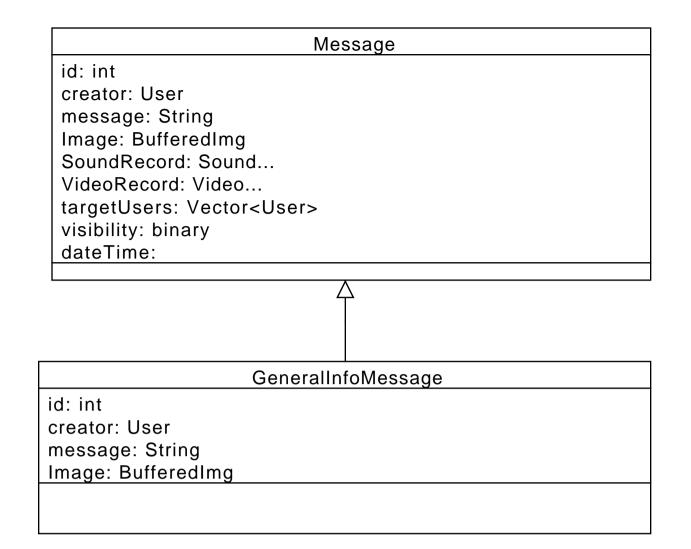
author: Vector<int>
moderator: Vector<int>
permission: binary

get y sets correspondientes
Vector<int> isAuthor()
Vector <int> isModerator()
setModerator(int idTablon)
setAuthor(int idTablon)

AnonymousUser
id: int
name: string
get y set correspondiente

La figura del dueño del tablón se va a distinguir, con un método en la clase user isAuthor() que devuelve un array de ids de tablon los cuales es autor. setAuthor indica que ese usuario es el autor de dicho tablon

Al igual con el autor, se puede saber que un usuario es moderador de un/os tablon/es isModerator() devuelve un array de enteros que indican el identificador del tablon del cual ese usuario modera. setModerator indica que ese usuario es el moderador de dicho tablon



# dataBaseManager

boolean checkUser(User user)
tablon getTablon(int tablonId)
message getMessage(int tablonId,
Message msgId)
addMessageToTablon(int tablonId,Message msg)
createTablon(Tablon tablon);
deleteTablon(int tablonId);
modifyTablon(int idTablon,Tablon tablon);

Habría que pensar si es más rentable traer todo el tablón o ya filtramos en el servidor que 'tablón' es el que quiero (visibilidad usuario)